



# Tecnológico de Monterrey

## REPORTE FINAL DEL PROYECTO DEL LABORATORIO DE MECATRÓNICA

### Latte Art Printing Machine

Instructor: Hiram Uribe

- ❖ Jonhatan Josué Robledo Ríos      A00811938
- ❖ Jaime Morales Gómez              A00813499
- ❖ Jacob Moreno Orta                  A00811484

# Índice

|                                       |           |
|---------------------------------------|-----------|
| <b>Índice</b>                         | <b>2</b>  |
| <b>Descripción General</b>            | <b>3</b>  |
| <b>Sistema Mecánico</b>               | <b>6</b>  |
| Componentes impresos                  | 6         |
| Base                                  | 7         |
| Poste para los rieles                 | 7         |
| Base para el riel                     | 8         |
| Mecanismo X                           | 9         |
| Mecanismo Y                           | 10        |
| Rieles                                | 11        |
| Base para la taza                     | 12        |
| <b>Sistema de control</b>             | <b>13</b> |
| Software utilizado                    | 13        |
| GRBL                                  | 13        |
| LaserGRBL                             | 14        |
| Sistema de control de motores a pasos | 15        |
| <b>Sistema electrónico</b>            | <b>16</b> |
| Diagrama esquemático de conexiones    | 16        |
| <b>Bibliografía</b>                   | <b>17</b> |
| <b>Anexos</b>                         | <b>17</b> |
| Código de Arduino de driver del motor | 17        |
| Imágenes del resultado final          | 21        |

## Descripción General

Nuestro proyecto consta de una máquina para hacer dibujos en la espuma del café, nos inspiramos en un video que encontramos en youtube y decidimos trabajar en él.



Al momento de analizar los aspectos que debía abarcar nuestro proyecto, nos dimos cuenta que la parte más retadora era la de la inyección de tinta, así que nos pusimos a investigar y optamos por seguir un tutorial que encontramos en internet:

<https://www.thingiverse.com/thing:2633175>

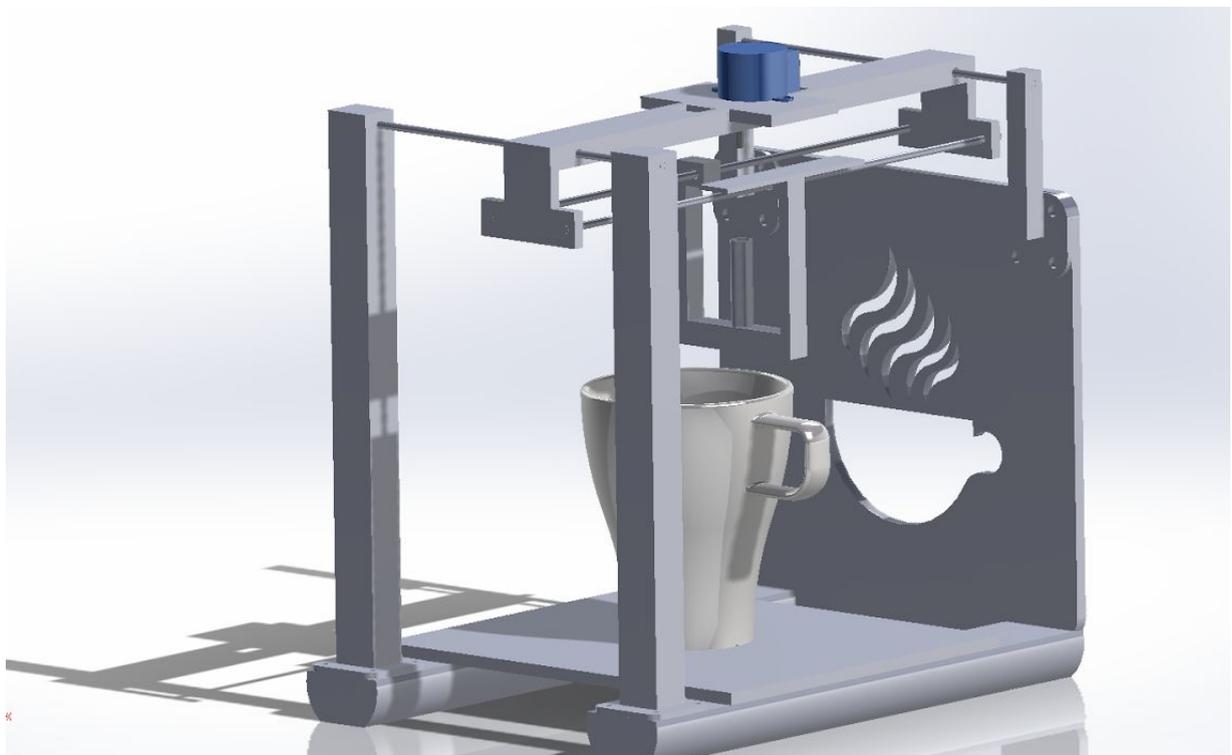
La inyección de tinta se basa en un mecanismo de una jeringa dosificadora, la cual está conectada a un motor de pasos que al moverse, hace que el mecanismo avance y empuje la jeringa para que pueda inyectar la tinta.

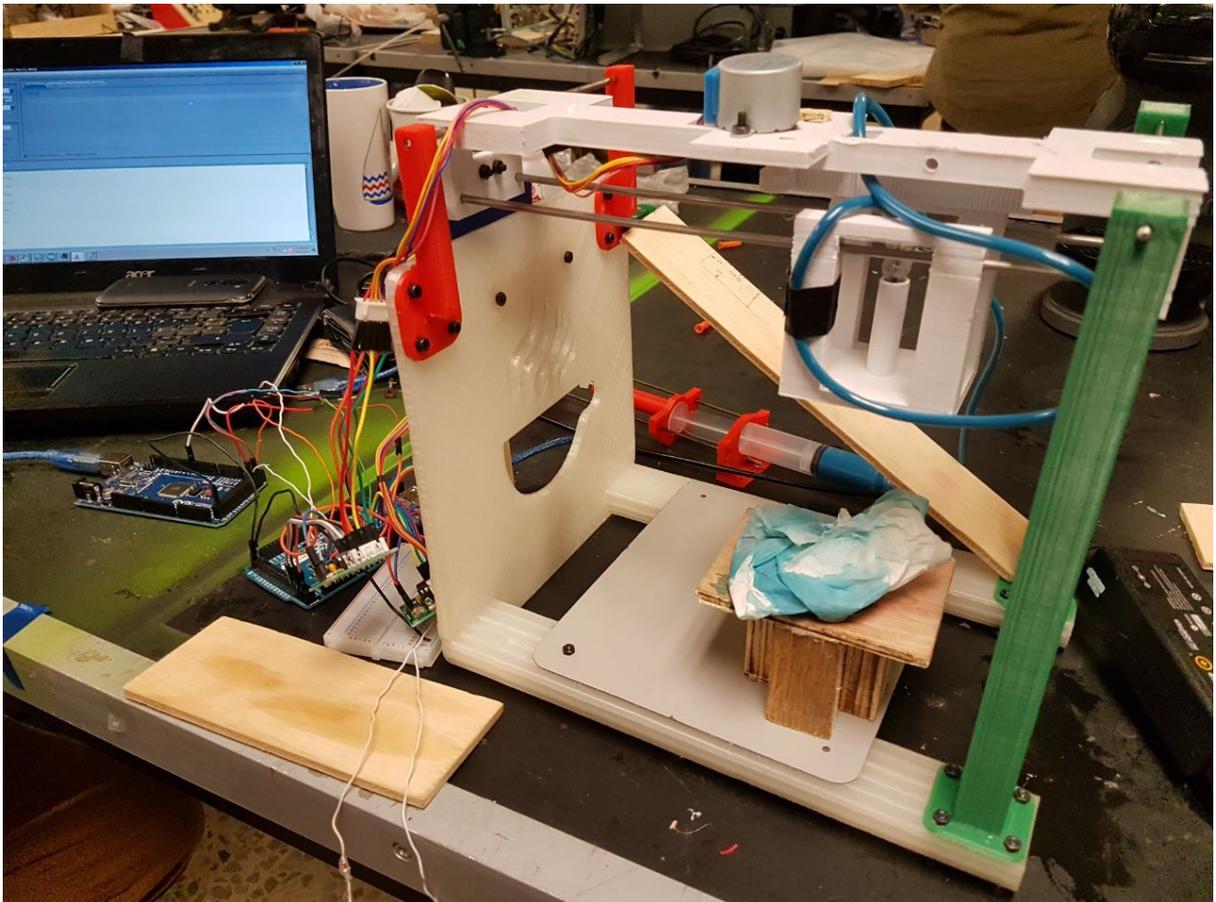
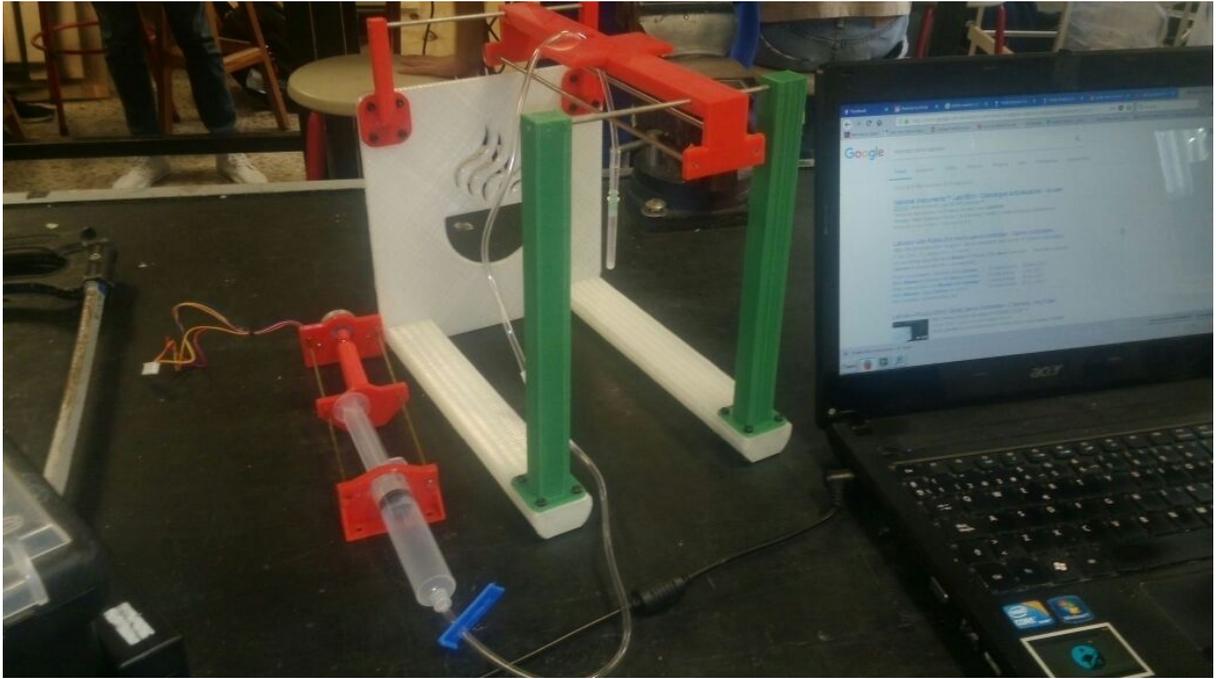
Para la parte mecánica realizamos el diseño de los componentes desde cero utilizando softwares como NX y SolidWorks, dichos componentes los imprimimos en 3D utilizando las impresoras del laboratorio de mecatrónica.

La parte electrónica y de control se trabajó mediante Arduino y el software GRBL para poder convertir cualquier dibujo que se quiera imprimir a un código G como el utilizado en las CNC y de esta manera lograr la impresión de las imágenes en el café.

Es un proyecto que no requirió mucha inversión de dinero ya que todos los componentes mecánicos que se necesitaron fueron impresos en 3D, lo cual es gratis dentro del Tec para los alumnos de los laboratorios de mecatrónica. Lo único que tuvimos que comprar fue el arduino para la parte de programación y control, los tres motores que se necesitaron y los tornillos, guasas y tuercas para el ensamblaje de las piezas. La inversión resultó de aproximadamente \$600.

A lo largo de este documento se irá explicando más a detalle cada uno de los sistemas con los que cuenta nuestro proyecto. A continuación se muestran imágenes de nuestra impresora de latte, tanto del diseño como de la máquina real.





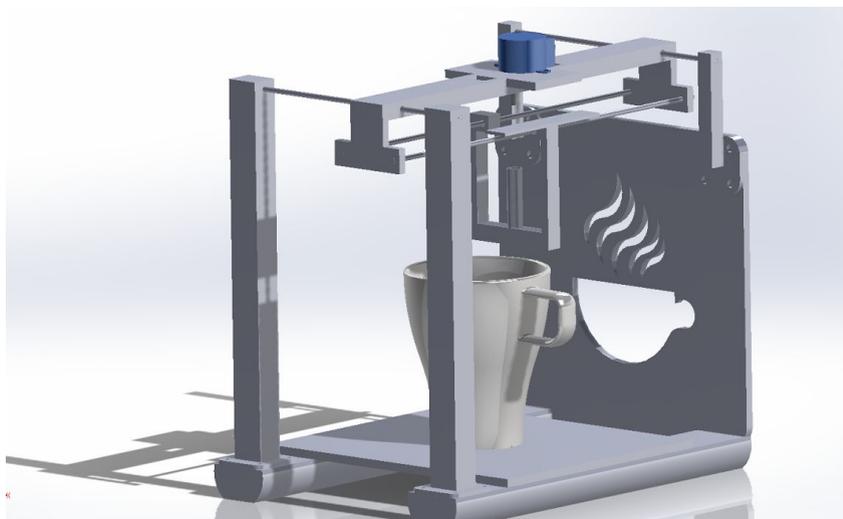
# Sistema Mecánico

La mayor parte de las piezas que se utilizaron para el sistema mecánico fueron impresas en 3D, pero a continuación se muestra una lista de los componentes que compramos.

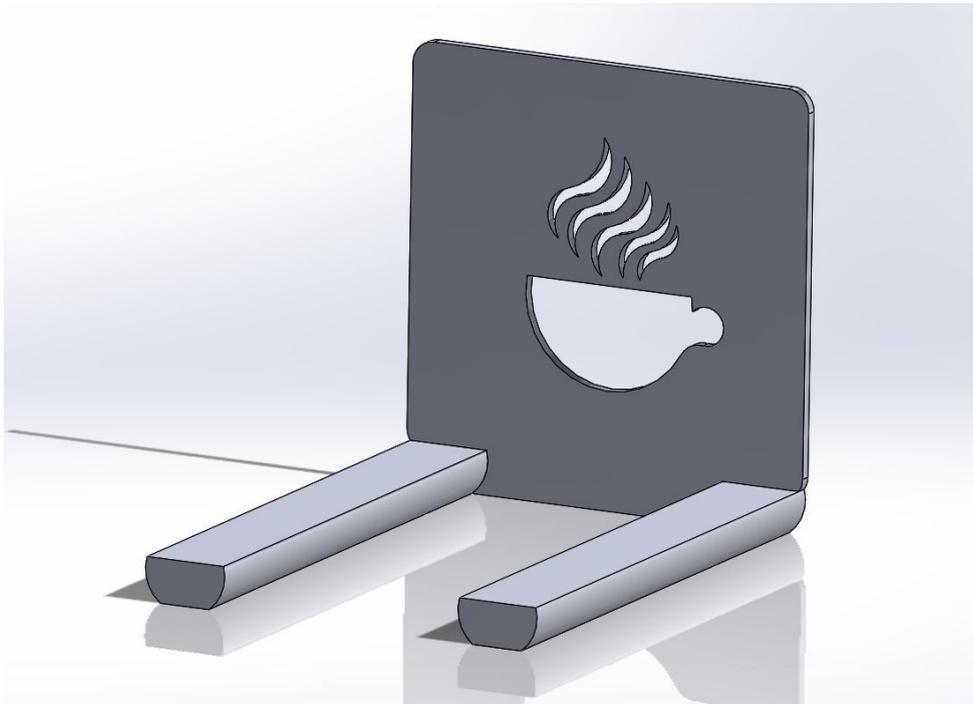
| Cantidad | Descripción                                  | Costo unitario | Link del proveedor  |
|----------|--|----------------|---|
| 3        | Motor a pasos con controlador 28BYJ-48 de 5V | \$55           | <a href="https://troxino.com/">https://troxino.com/</a>     |
| 10       | Tornillos m3 de 25 mm                        | \$1.62         | <a href="http://www.bytnsa.com/">http://www.bytnsa.com/</a> |
| 14       | Tornillos m3 de 12 mm                        | \$1.52         | <a href="http://www.bytnsa.com/">http://www.bytnsa.com/</a> |
| 24       | Tuercas m3                                   | \$0.20         | <a href="http://www.bytnsa.com/">http://www.bytnsa.com/</a> |

## Componentes impresos

A continuación se explicará a detalle cada uno de los componentes que forman parte de la máquina impresora de Latte.

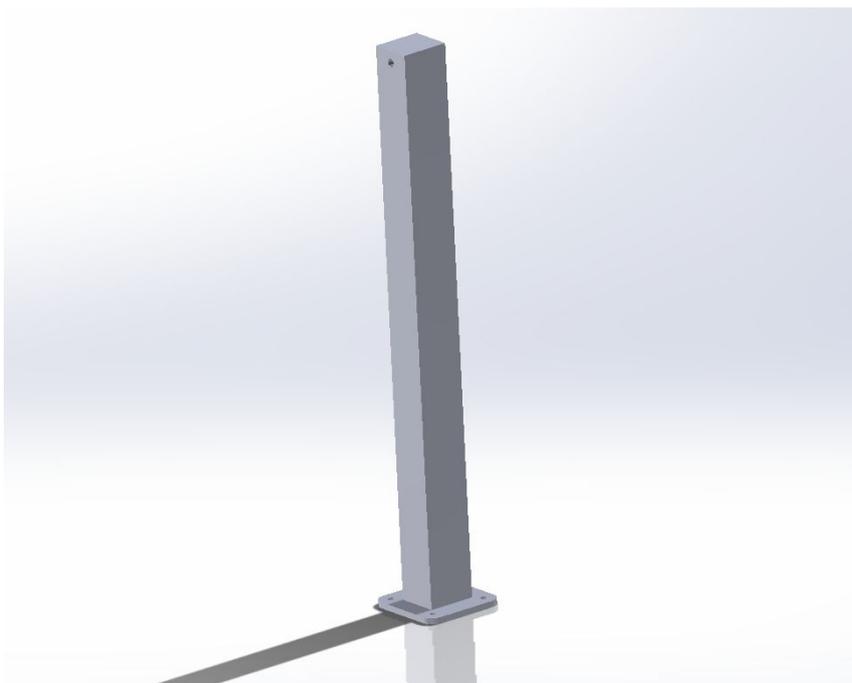


## Base



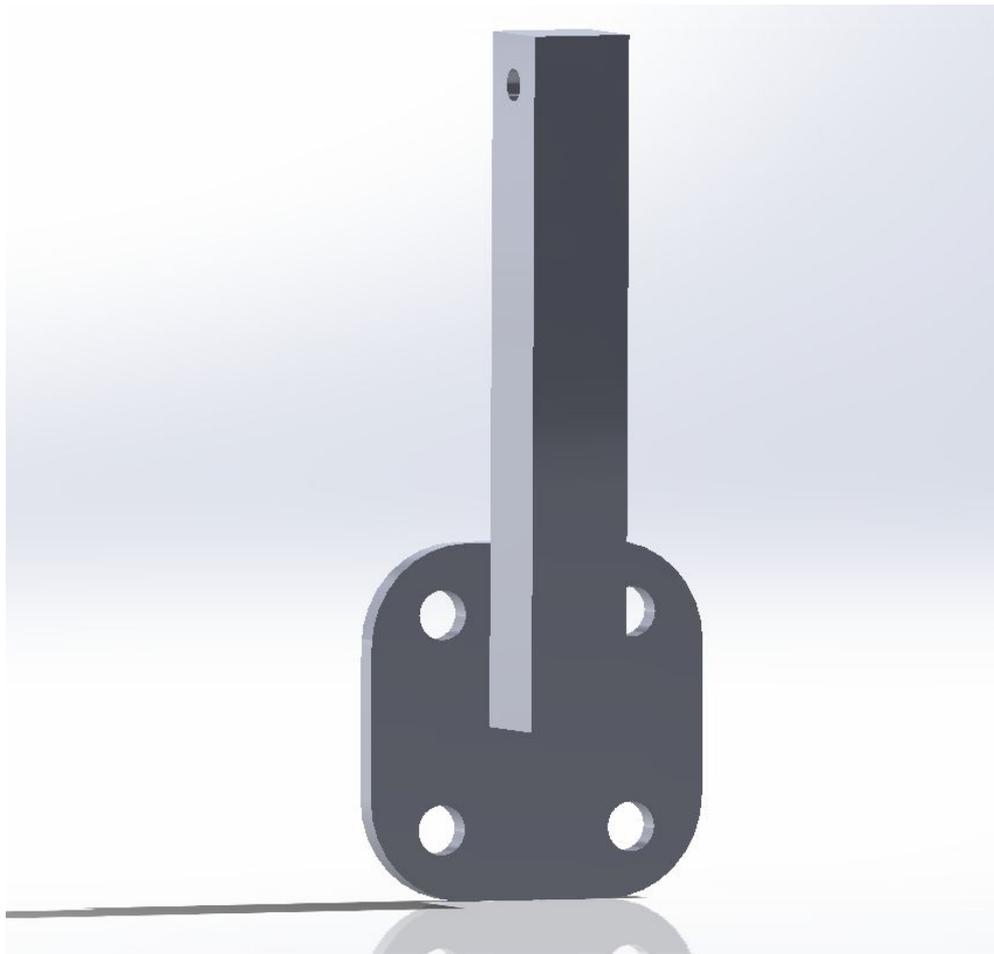
Esta pieza es la base de toda la estructura, consta de dos “patas” y una pared con un diseño de una taza de café inspirado en el logotipo del Tec de Monterrey.

## Poste para los rieles



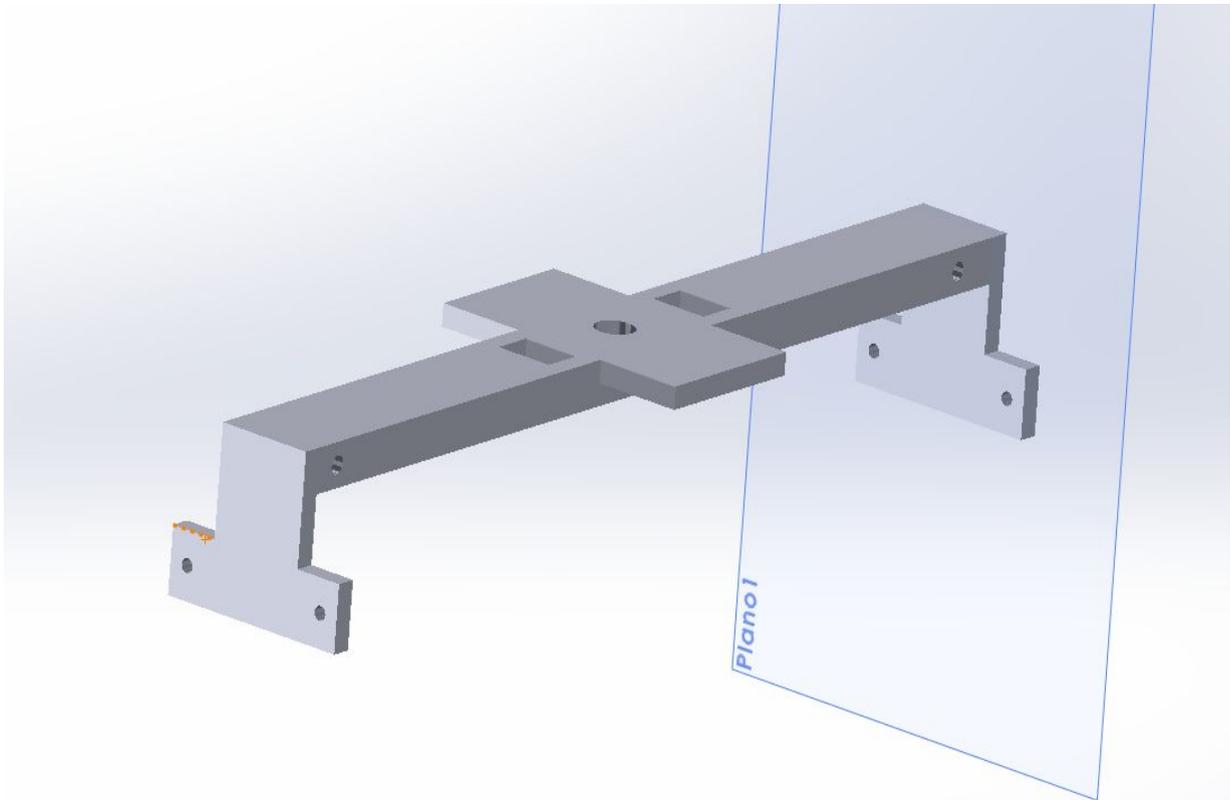
Este poste es el encargado de sostener los rieles por donde se va a mover nuestro mecanismo en el eje X. Se necesitan dos de estos postes. Es importante resaltar que hay que tomar en cuenta los agujeros por donde se van a insertar los tornillos, en este caso utilizamos tornillos de 3 mm.

### Base para el riel



Se necesitan dos de estas bases, son las que tienen la función de sostener el otro riel del eje X. Se encuentran atornilladas a la pared también con tornillos de 3mm.

## Mecanismo X



Este mecanismo es el encargado de que la jeringa se mueva en el eje X. En la parte de arriba se monta el motor, es por eso que tiene un agujero circular para la flecha, y dos agujeros rectangulares para que puedan pasar los cables del motor por ahí.

Podemos observar que además cuenta con otros 6 agujeros; los dos de la parte de arriba son por donde pasan los rieles del eje X, mientras que los 4 de abajo servirán para montar los rieles para el eje Y.

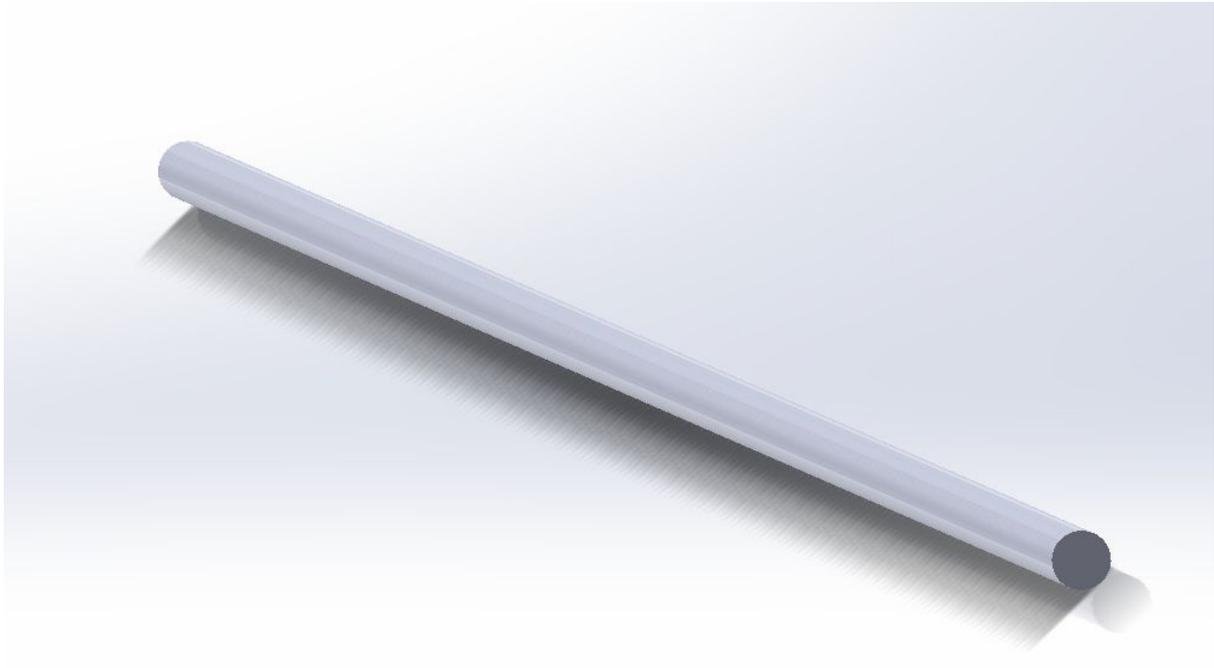
## Mecanismo Y



Este mecanismo es el encargado de que la jeringa se mueva en el eje Y. El cilindro central es la parte en donde se coloca la jeringa, mientras que el rectángulo que se encuentra en la parte de arriba sirve como base para la cremallera.

Es importante resaltar que tanto el mecanismo X como el mecanismo Y se mueven por medio de un sistema de cremallera-piñón conectado a un motor de pasos. Los agujeros que tienen las piezas funcionan como bujes para que pasen los rieles, pero se presentaron algunos problemas en el movimiento sobre todo con el mecanismo X, ya que la separación entre los rieles es muy larga, entonces nuestra recomendación es considerar la implementación de baleros para permitir un mejor movimiento.

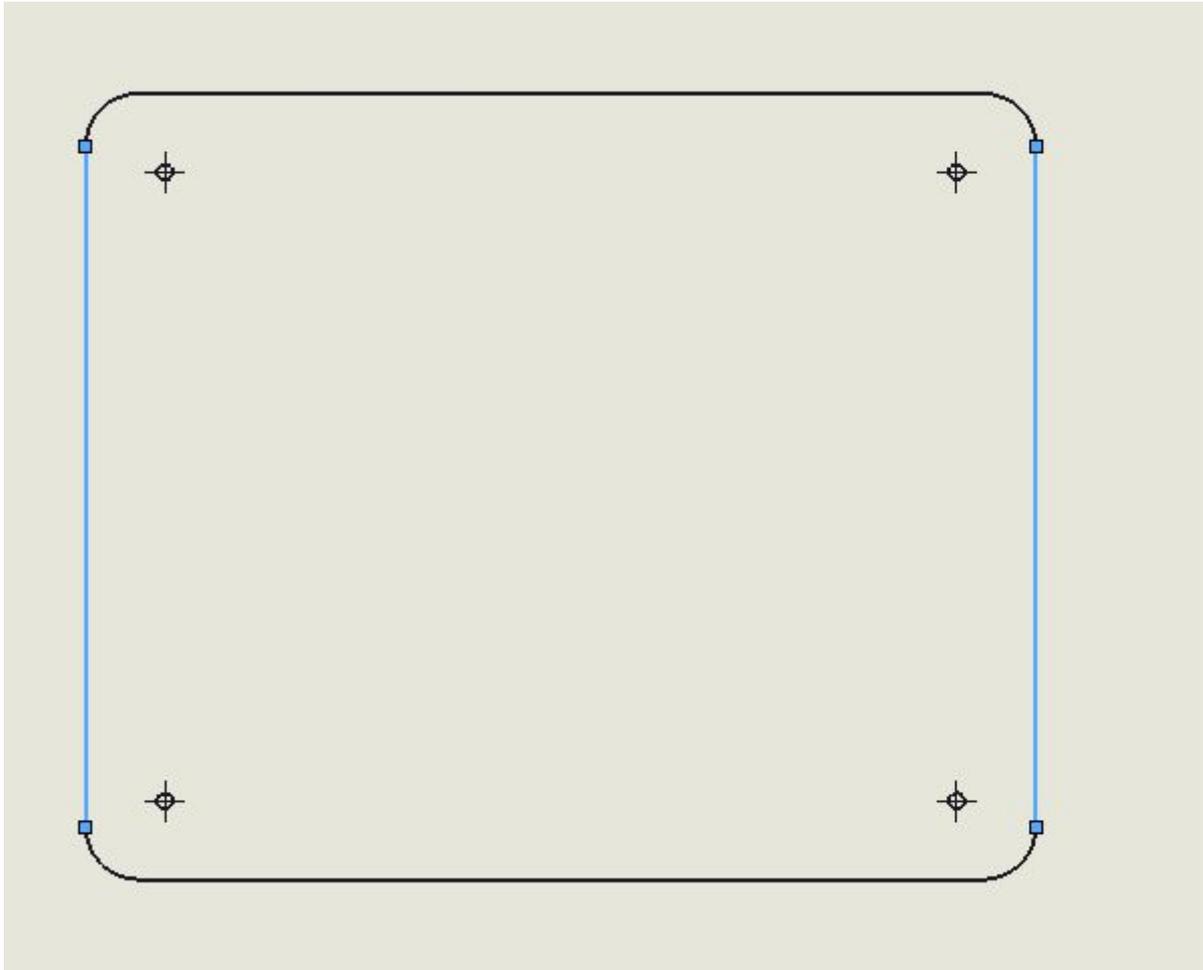
## Rieles



Para los rieles utilizamos cilindros de acero de 3.1 mm, decidimos utilizar este tamaño porque eran los que había en el laboratorio y así no teníamos que comprar. Además, como nuestra máquina iba a ser relativamente pequeña, no se necesitaban rieles más grandes.

Nosotros recomendamos utilizar rieles más grandes y comprar baleros para no tener problemas con el movimiento de los mecanismos. Como ya mencionamos antes, tuvimos algunos problemas para el movimiento del eje X, así que para evitar conflictos de este estilo, ya sea por fricción o porque los agujeros de las piezas dejan mucho juego para que se mueva el riel, nuestra recomendación es implementar un sistema con baleros.

## Base para la taza



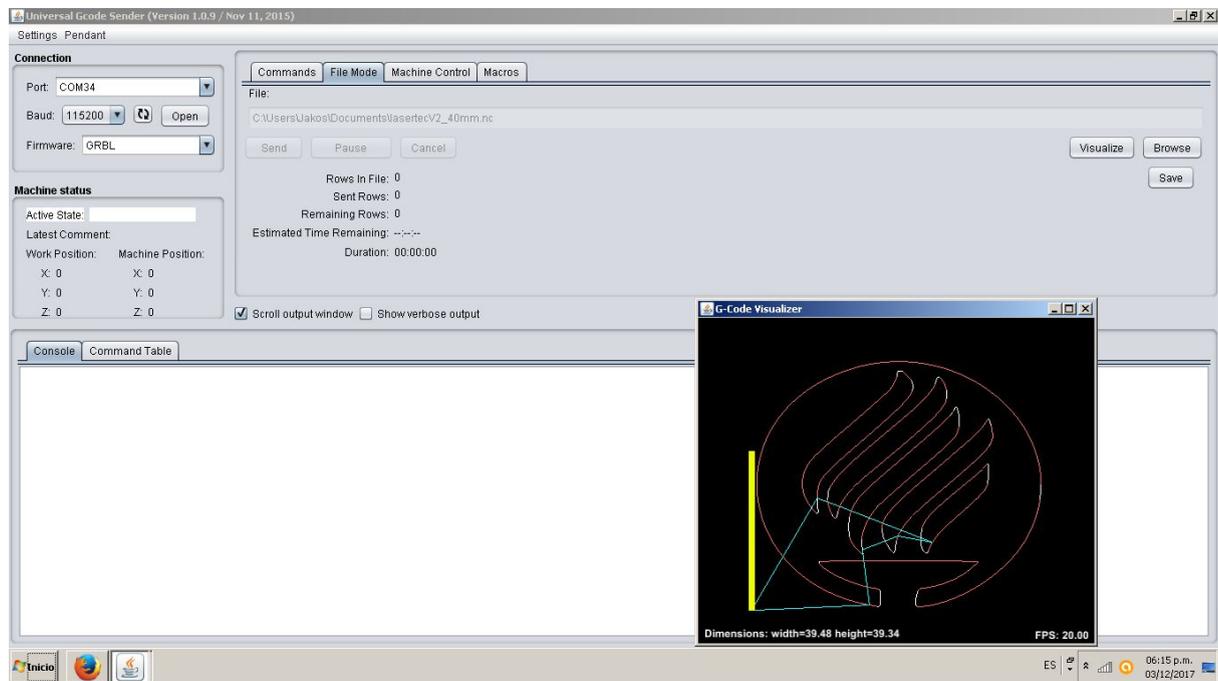
Para realizar la base en donde se encuentra montada la taza, utilizamos la cortadora de agua que se encuentra en el campus en el sótano de cetec. Utilizamos la carcasa de un DVD que ya no servía y lo cortamos con una “Waterjet”.

No se imprimió en 3D por dos cosas: es mucho más rápido el corte de agua y además las dimensiones excedían el permitido por una impresora 3D.

# Sistema de control

## Software utilizado

### GRBL



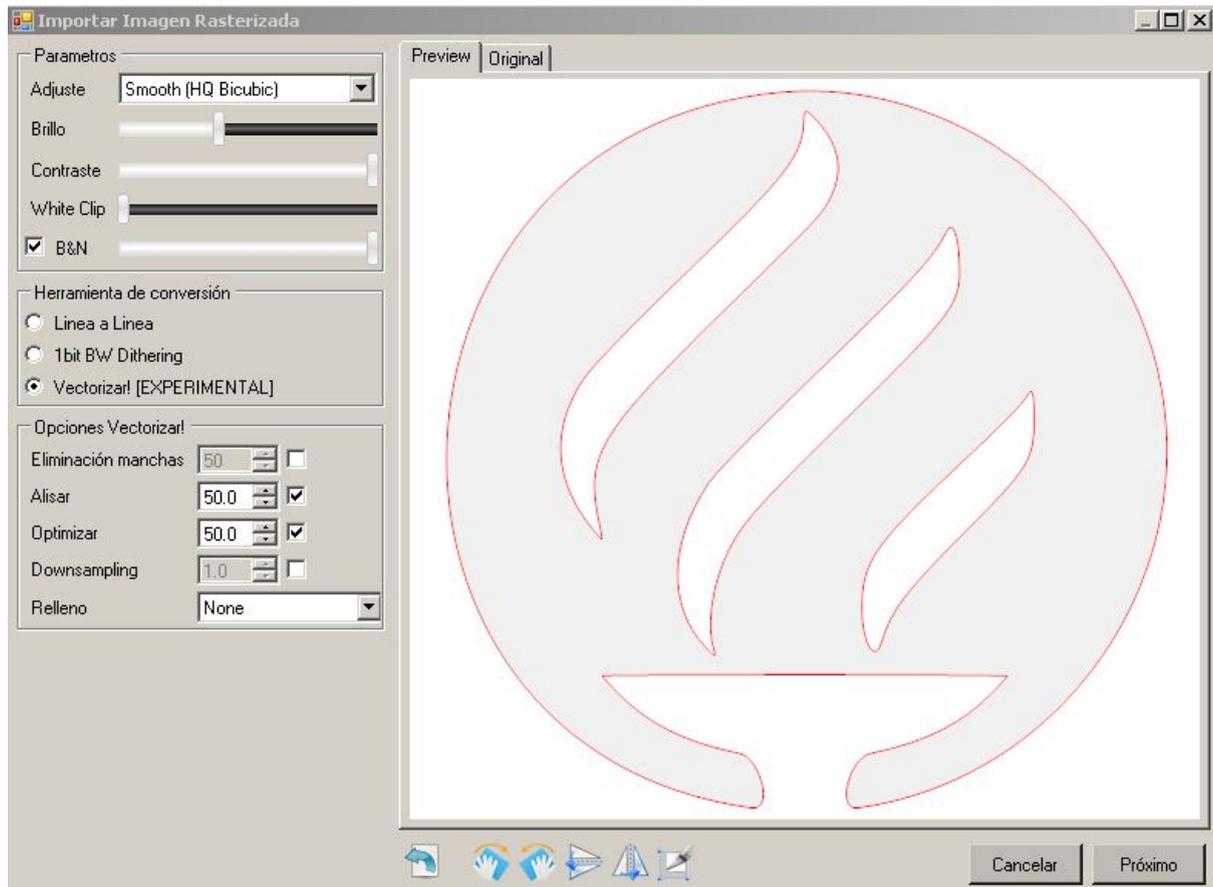
Para el control de la impresora se utilizó el software de uso libre GRBL en una versión modificada para su uso en la tarjeta arduino Mega 2560. Este es una librería de arduino diseñada para dar soporte a routers CNC. con capacidades de interpretación de código G , control de motores de pasos en 3 ejes cartesianos, cambios de velocidad de giro del husillo, entre otros. Estas 3 funcionalidades fueron las principales utilizadas para el proceso de control de la impresora ya que se utilizó 2 ejes cartesianos para controlar el plano de impresión, y el tercero en conjunto con el activado y desactivado del movimiento del husillo para el control del actuador utilizado para deposición de tinta.

Para comunicación con el GRBL se utilizó el **Universal Gcode sender**.

Su interfaz Se compone de una consola de comandos para escribir directamente código G o comandos del sistema GRBL, una pestaña para carga de archivos de programas en código G, un visualizador de las trayectorias que simula el movimiento de la herramienta, una pestaña

para control manual del sistema y una pestaña para configuración de macros para rápido acceso.

## LaserGRBL



Para el convertidor de imágenes a código G utilizable por nuestro sistema, se utilizó el software LaserGRBL, que permite vectorizado de imágenes y conversión a código G con soporte para grabado con láser, por lo que las salidas pueden ser adaptadas para utilización con la cabeza de impresión de la siguiente manera:

El programa genera trayectorias a través de curvas y rectas con una velocidad de feed definida como lo haría cualquier router, pero este utiliza códigos M3 y M5 para encender o apagar el laser como lo haría con el Husillo. El GRBL utiliza uno de los pines como pin de encendido y apagado del husillo por lo que este mismo puede ser reutilizado para activar o desactivar la deposición de tinta.

## Sistema de control de motores a pasos

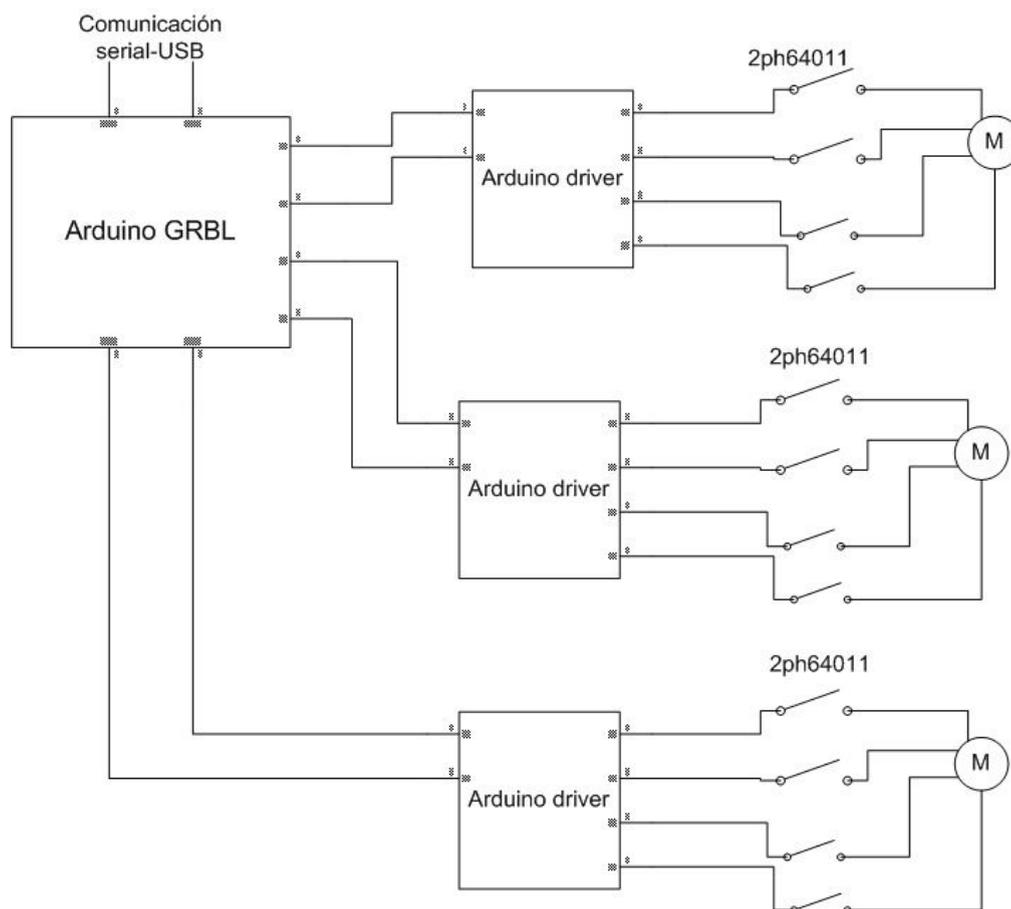


Para control de los motores 28byj-48 se utilizaron drivers 2ph64011a que a través de 4 pines de entrada, dictan las bobinas del motor a pasos a energizar. Dado que el GRBL entrega una señal de control en 2 pines (pasos, dirección) se utilizó un segundo arduino a modo de driver que manejara a través de interrupciones los cambios en el estado de cada motor, así como un manejo especial de la cabeza de impresión dado que se requería solo del control de los estados activado y desactivado.

# Sistema electrónico

| Cantidad | Descripción                        | Precio Unitario                             | Link del proveedor  |
|----------|------------------------------------|---|---|
| 2        | Arduino UNO                        | \$215                                       | <a href="https://www.lionchipmexico.com/">https://www.lionchipmexico.com/</a> |
| 3        | Driver para motor de paso 28byj-48 | Incluido con el motor (\$55 motor y driver) | <a href="https://troxino.com/">https://troxino.com/</a>                       |

## Diagrama esquemático de conexiones



# Bibliografía

- L. (2014, June 10). LETARTARE/Grbl-xx\_with\_Arduino. Retrieved December 03, 2017, from [https://github.com/LETARTARE/Grbl-xx\\_with\\_Arduino/tree/devArduino](https://github.com/LETARTARE/Grbl-xx_with_Arduino/tree/devArduino)
- T. (n.d.). Syringe dosing unit by kpr. Retrieved December 03, 2017, from <https://www.thingiverse.com/thing:2633175>
- Bass guitar. (n.d.). Retrieved December 03, 2017, from <https://openthings.wiki/index>

# Anexos

## Código de Arduino de driver del motor

```
// Definimos los pines donde tenemos conectadas las bobinas
//diseñado para insertar las tarjetas en los pines y solo conectar con cables a la fuente de 5v
```

```
#define INink1 8
#define INink2 9
#define INink3 10
#define INink4 11
```

```
#define INy1 23
#define INy2 25
#define INy3 27
#define INy4 29
```

```
#define INx1 4
#define INx2 5
#define INx3 6
#define INx4 7
```

```
//define pines driver
#define Dir_1 1
#define Dir_2 2
#define Dir_tinta_3 3
```

```

#define steps_1 18
#define steps_2 19
#define steps_Tinta_3 20

int stepDir_1;
int stepDir_2;
int stepDir_tinta_3;
int i1=0;
int i2=0;
int i3=0;

// Secuencia de pasos (par máximo)
int paso [16][4] =
{
  {1, 0, 0, 0},
  {1, 1, 0, 0},
  {0, 1, 0, 0},
  {0, 1, 1, 0},
  {0, 0, 1, 0},
  {0, 0, 1, 1},
  {0, 0, 0, 1},
  {1, 0, 0, 1},

  {0, 0, 0, 1},
  {0, 0, 1, 1},
  {0, 0, 1, 0},
  {0, 1, 1, 0},
  {0, 1, 0, 0},
  {1, 1, 0, 0},
  {1, 0, 0, 0},
  {1, 0, 0, 1}

};
unsigned long timelaststep=0;
void setup()
{
  attachInterrupt(digitalPinToInterrupt(steps_1), stepYdetected, FALLING);
  attachInterrupt(digitalPinToInterrupt(steps_2), stepXdetected, FALLING);
  //attachInterrupt(digitalPinToInterrupt(steps_Tinta_3), stepINKdetected, FALLING);
  // Todos los pines en modo salida
  pinMode(INx1, OUTPUT);
  pinMode(INx2, OUTPUT);
  pinMode(INx3, OUTPUT);
  pinMode(INx4, OUTPUT);
  pinMode(INy1, OUTPUT);

```

```
pinMode(INy2, OUTPUT);
pinMode(INy3, OUTPUT);
pinMode(INy4, OUTPUT);
pinMode(INink1, OUTPUT);
pinMode(INink2, OUTPUT);
pinMode(INink3, OUTPUT);
pinMode(INink4, OUTPUT);
```

```
pinMode(Dir_1, INPUT);
pinMode(Dir_2, INPUT);
pinMode(Dir_tinta_3, INPUT);
pinMode(steps_1, INPUT);
pinMode(steps_2, INPUT);
pinMode(steps_Tinta_3, INPUT);
}
```

```
void loop(){
```

```
if(digitalRead(steps_Tinta_3)){
```

```
    //activa laser con pin 6 grbl mega
    if((timelaststep-millis())>20){
    if(digitalRead(Dir_tinta_3)){
        stepDir_tinta_3=15-i3;
    }
    else{
        stepDir_tinta_3=7-i3;
    }

```

```
        digitalWrite(INink1, paso[stepDir_tinta_3][0]);
        digitalWrite(INink2, paso[stepDir_tinta_3][1]);
        digitalWrite(INink3, paso[stepDir_tinta_3][2]);
        digitalWrite(INink4, paso[stepDir_tinta_3][3]);
        timelaststep=millis();
        if(i3<8){
            i3++;
        }else{
            i3=0;
        }
    }//digitalin
}//micros
}
```

```

void stepYdetected(){
  if(digitalRead(Dir_1)){
    stepDir_1=15-i1;
  }
  else{
    stepDir_1=7-i1;
  }

  digitalWrite(INy1, paso[stepDir_1][0]);
  digitalWrite(INy2, paso[stepDir_1][1]);
  digitalWrite(INy3, paso[stepDir_1][2]);
  digitalWrite(INy4, paso[stepDir_1][3]);
  if(i1<8){
    i1++;
  }else{
    i1=0;
  }
}

```

```

}
void stepXdetected(){
  if(digitalRead(Dir_2)){
    stepDir_2=15-i2;
  }
  else{
    stepDir_2=7-i2;
  }

  digitalWrite(INx1, paso[stepDir_2][0]);
  digitalWrite(INx2, paso[stepDir_2][1]);
  digitalWrite(INx3, paso[stepDir_2][2]);
  digitalWrite(INx4, paso[stepDir_2][3]);
  if(i2<8){
    i2++;
  }else{
    i2=0;
  }
}

```

```

}
void stepINKdetected(){

```

```

}

```

Imágenes del resultado final



